REMARKS

The present amendment and remarks are in response to the final office action entered in the above-identified case and mailed on September 14, 2007. On October 15, 2007, Applicants filed a Notice of Appeal along with a Pre-Appeal Brief Request for Review. A Notice of Panel Decision from the Pre-Appeal Brief Review indicating the panel's decision to proceed to the Board of Patent Appeals and Interferences was mailed on December 14, 2007. The Notice of Panel Decision reset the period for filing an Appeal Brief to one month from the mailing date of the decision. Further, the period for filing the Appeal Brief is extendable under 37 C.F.R. 1.136. Thus, the present response, accompanied by an RCE, a request for a five month of extension of time, and the corresponding fee, is timely filed.

Claims 1-79 are pending in the application. Claims 1, 15-18, 31-46, 48-50 and 53-55 stand rejected under 35 U.S.C. §102(b) as being anticipated by U.S. Patent No. 6,954,724 to Kodosky, et al. (hereafter Kodosky). Claims 2-5, 9, 19-22, 26, 47, 58-74 and 78 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Kodosky in view of U.S. Patent Publication No. 2002/0194218 to Klapper, et al. (hereafter Klapper). Claims 12-14, 28-30 and 52 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Kodosky in view of U.S. Patent No. 5,845,063 to Khranpunovich, et al. (hereafter Khrapunovich). Claims 6-8, 23-25, 77 and 79 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Kodosky in view of Klapper and further in view of Khrapunovhch. Finally, claims 10, 11 and 27 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Kodosky in view of Klapper, further in view of U.S. Patent No. 6,369,836 to Larson, et al. (hereafter Larson). Applicants have amended the claims as indicated in the attached listing of claims.

With these changes, Applicants respectfully submit that all of the pending claims are now in condition for allowance.

Rejections Under 35 U.S.C. § 102(b)

Turning first to the rejection of claims 1, 15-18, 31-46, 48-50 and 53-55 under 35 U.S.C. § 103 over Kodosky, applicants note that claims 1, 18 and 34 are independent claims. Each of these claims has been amended to more clearly distinguish the invention over the art of record. With these changes 1, 18 and 34, as well as the claims depending therefrom, are not anticipated by Kodosky and should be allowed.

A claim is anticipated under 35 U.S.C. § 102(b) only if every element of the claim is found in a single prior art reference. Amended claims 1, 18 and 34 include a number of features that are not disclosed in Kodosky. Claim 1, for example, calls for a method of configuring a state machine implemented in a function block associated with a process plant. The state machine implemented within the function block defines a plurality of states. The state machine transitions between states based on state machine configuration data and one or more state machine inputs associated with operation of the process plant. The method of configuring the state machine calls for providing a graphical user interface which is displayed by a display device. The graphical user interface includes a plurality of graphical elements that define input/state pairs. The method of claim 1 calls for receiving state transition data associated with one or more of the plurality of graphical elements. The state transition data identify next states associated with the graphical elements. A state machine next state associated with a particular graphical element identifies the next state to which the state machine transitions when conditions in the process plant correspond to the input/state pair

defined by the graphical element. For example, the state machine will transition to the next state associated with a graphical element when the state machine is in the state defined by a particular graphical element input/state pair and when the input defined by the graphical element input/state pair is a certain value. Finally, the method of claim 1 calls for storing the state transition data on a first computer readable medium associated with the function block implementing the state machine.

Kodosky does not teach each and every element of claim 1. In fact, Kodosky does not teach a method of configuring a state machine at all. Rather, Kodosky teaches a system and method for converting a graphical program into a programmable hardware implementation. (Col. 1, lines 31-35). The computer implemented system of Kodosky automatically generates hardware level functionality, *e.g.*, programmable hardware or FPGAs, in response to a graphical program created by a user. This allows the user to define instrument functionality using graphical programming techniques while enabling the resulting program to operate directly in hardware. (Col. 4, lines 16-23). The user creates a graphical program that performs the desired functionality, and the graphical program is converted into an executable form, with at least a portion of the graphical program converted into a hardware implementation. (Col. 4, lines 31-34). The portion of the graphical program selected for hardware implementation is exported to a hardware description. The hardware description is converted to a Net List, preferably an FPGA-specific Net List. The Net List is compiled into an FPGA program file or software bit stream. (Col. 4, lines 46-54). The resulting bit stream is transferred to an FPGA device to produce a programmed FPGA equivalent to the graphical program. (Col. 4, lines 65-67).

Nothing in the Kodosky disclosure has anything to do with configuring a state machine implemented in a function block associated with a process plant. The only mention of a state machine in the entire Kodosky disclosure occurs in the description of Fig. 14. Fig. 14 is a state diagram illustrating the operation of a While Loop function block that may be implemented in a programmable hardware device in Fig. 13. (Col. 19, lines 49-51). Fig. 14 merely illustrates the various states the "state machine" (*i.e.*, the while loop function block) may take on and the transitions between the various states as the operation of the While Loop proceeds. (See generally Fig. 14, col. 19, line 48 – col. 20, line 19). The description of Fig. 14 does not describe how the While Loop is configured, how the various states are defined, or how the transitions between the states are configured.

Thus, Kodosky does not teach providing a graphical user interface including a plurality of graphical elements defining state machine input/state pairs. Kodosky does not teach receiving state transition data associated with one or more graphical elements. Kodosky does not teach receiving state transition data identifying one or more next states to which a state machine is to transition following conditions in a process plant that correspond to input/state pairs defined by one or more graphical elements, in fact, Kodosky does not teach receiving state transition data of any kind. Finally, Kodosky does not teach storing state transition data on a first computer readable medium associated with a function block implementing a state machine. Absent a teaching of these claimed elements Kodosky cannot anticipate independent claim 1 of the present application, nor the claims depending therefrom.

Claims 18 and 34 are similar to claim 1. Claim 18 calls for a tangible medium storing machine readable code comprising first code, second code, and third code. The first code provides a graphical user interface via a display device for configuring state machine transitions among a plurality of state machine states. The graphical user interface includes a plurality of graphical elements representing state machine input/state pairs which can be used to indicate desired transitions between states. The second code receives state transition data identifying a state machine next stage associated with one of the graphical elements via the graphical user interface. The third code stores the state transition data on a computer readable medium associated with the function block implementing the state machine. The state machine transitions to the next state when conditions in the process plant correspond to the input/state pair associated with the graphical element.

Claim 34 calls for a method of implementing a state machine in a function block for use in controlling or simulating the control of one or more field devices in a process plant. As with claim 1, the method of claim 34 calls for providing a graphical user interface displayed by a display device. The graphical user interface includes a plurality of graphical elements for configuring state machine transitions between a plurality of state machine states. The graphical elements define one or more input/state pairs which are indicative of one or more conditions in a process plant. The method next calls for receiving state transition data identifying a state machine next state associated with at least one of the graphical elements. The state transition data are then stored on a first computer readable medium associated with the function block. The method of implementing a state machine then calls for receiving at least one state machine input, determining a state machine next state based on the at least one input, the current state, and the state transition data. The current state is then set to the next

state, and an output is provided by the function block for controlling or simulating the control of the one or more field devices in the process plant based on the current state.

As with claim 1, Kodosky does not teach all of the elements of amended claims 18 and 34. Kodosky teaches a graphical user interface, but it is not used for configuring the transitions of a state machine between a plurality of states. Rather, Kodosky's graphical user interface is used for creating a graphical program, a portion of which will be translated into hardware level functionality in a programmable device such as an FPGA. Kodosky's graphical user interface does not include a plurality of graphical elements representing state machine input/state pairs as called for in both claims 18 and 34 of the present application. Nor does Kodosky teach code for, or the step of, receiving state transition data identifying a state machine next state associated with one of the graphical elements. Further, since Kodosky does not teach receiving such information, it follows that Kodosky cannot teach storing such transition data. Again, absent a teaching of these claimed features, Kodosky cannot anticipate independent claims 18 and 34 and the claims depending therefrom.

Rejections Under 35 U.S.C. § 103(a)

Next applicants turn to the rejection of claims 2-5, 9, 19-22, 26, 47, 58-74, 76 and 78 under 35 U.S.C. § 103(a) as being unpatentable over Kodosky in view of Klapper. Applicants first address the rejection of claim 58 over Kodosky and Klapper, as claim 58 is the only independent claim rejected under 35 U.S.C. § 103(a) over Kodosky and Klapper. It is well settled that a claim in a patent application is unpatentably obvious over a combination of prior art references only if the combined references teach or suggest every element of the

invention claimed. In Re Royka, 490 F.2d 981, 180 U.S.P.Q. 580 (CCPA 1974). In the present case, claim 58 is not unpatentable over the combined teaching of Kodosky and Klapper because, even when combined, Kodosky and Klapper do not teach or suggest every element of the invention claimed in claim 58.

Claim 58 claims a function block entity implemented in a process plant for controlling, or simulating the control of, one or more field devices. Claim 58 calls for a user modifiable state machine database. The state machine database includes state transition data indicative of the manner in which the state machine implemented by the function block is to transition among a plurality of different states. The transition data include pairings of states and function block inputs such that the transition data identifies the next state that the state machine transitions to when the state machine is in a state corresponding to one of the state/input pairings and when the input identified in the state/input pairing is a particular value. Computer code associated with the function block is adapted to receive function block inputs associated with the operation of the process plant, and determine the state machine next state based on at least one of the function block inputs, the current state of the state machine, and the state transition data. Further computer code causes the state machine to transition to the identified next state and sets one or more outputs for controlling the field device based on the current state of the state machine.

The examiner acknowledges that Kodosky does not each a user modifiable state machine configuration database (O.A. page 17). For this the examiner relies on Klapper. The examiner states that Klapper teaches a user modifiable matrix database stored on a memory. According to the examiner, it would have been obvious to one of ordinary

skill in the art to combine the user defined parameters of a matrix database of Klapper with the state machine taught by Kodosky to arrive at the invention claimed in claim 58. While Klapper does disclose a cause-and-effects matrix which allows a user to define matrix parameters, the examiner mischaracterizes Klapper's matrix as teaching the state machine configuration database called for in claim 58 of the present application. The state machine configuration database of claim 58 includes state transition data indicative of how the state machine implemented in the function block transitions among states. The state transition data stored in the database comprises state/input pairings and state transition data indicative of a next state to which the state machine is to transition when then state machine is in a state corresponding to one of the state/input pairings and the corresponding input of the state/input pairing is a particular value. Klapper's cause-and-effects matrix allows a user to define cause parameters, effects parameters, and intersection parameters. (See paragraphs 29-37). The cause-and-effects matrix defines input elements or variables that require monitoring, output responses to changes in the input elements/variables and the relationships between the input elements/variables and the output responses. (Paragraph 26). Klapper's cause-and-effects matrix teaches nothing regarding the storage of state transition data indicative of how a state machine implemented in a function block is to transition among a plurality of states as called for in claim 58. Furthermore, the cause parameters, effects parameters, and intersection parameters disclosed by Klapper do not comprise state/input pairings as called for in claim 58. At most, the cause parameters and effects parameters may comprise input/output pairs with the intersection parameters defining the relationship therebetween. The cause and effect parameters and the intersection parameters of Klapper's matrix have no relationship to the state transitions of a state machine implemented in function block as called for in claim

58 of the present application. Accordingly, it cannot be said that Klapper teaches a user modifiable state machine configuration database as called for in claim 58 of the present application. Thus, even when Klapper is combined with Kodosky the resulting combination still fails to teach all of the elements of claim 58. Therefore claim 58 is not unpatentable over Kodosky and Klapper and should be allowed. Claims 59-74, 76 and 78 all depend either directly or indirectly from claim 58 and are allowable over the combined teaching of Kodosky and Klapper for the same reasons.

With regard to claims 2-5, 9, 19-22, 26 and 47 rejected under 35 U.S.C. § 103(a) as being unpatentable over Kodosky in view of Klapper, applicants note that claims 2- 5 and claim 9 depend from independent claim 1, claims 19-22 and claim 26 depend from claim 18, and claim 47 depends from claim 34. As has already been argued with respect to claims 1, 18 and 34, Kodosky fails to teach or suggest a graphical user interface that includes a plurality of graphical elements defining state machine input/state pairs. Kodosky does not teach or suggest receiving state transition data associated with one or more of the graphical elements identifying one or more next states to which a state machine transitions following conditions in a process plant corresponding to one of the input/state pairs defined by one of the graphical elements, and Kodosky does not teach or suggest storing state transition data on a computer readable medium associated with a function block implementing a state machine. With regard to claims 2 and 19, Klapper is cited as teaching cells associated with pairings of graphical elements with at least one input that is indicative of a next state. The examiner points to the cells of Klapper's cause-and-effects matrix as teaching the cells corresponding to input/state pairs as called for in claims 2 and 19 of the present application. However, the cells of Klapper's cause-and-effects matrix do not

correspond to input/state pairs as called for in the claims. Rather, the cells in Klapper's cause-and-effects matrix correspond to input/output pairs. Furthermore, even if the cells of Klapper's cause-and-effects matrix did correspond to input/state pairs as called for in the claims, the combination of Kodosky and Klapper would still fail to teach the elements of the base claims listed above. Therefore claims 2 and 19 are not unpatentable over Kodosky and Klapper.

The situation is similar with regard to claims 3 and 20, 4 and 21, 5 and 22, and 9 and 26. Klapper does not teach or suggest the subject matter called for in the dependent claims, and even if it did, the combination of Kodosky and Klapper would still fail to teach or suggest every element of the independent base claims. Briefly, with regard to claims 3 and 20, Klapper is cited as teaching displaying state transition data in the cells corresponding to input/state pairs. Since Klapper does not disclose cells corresponding to input/state pairs, it cannot teach displaying state transition data in such cells. With regard to claims 4 and 21 Klapper is cited as teaching a matrix of cells wherein the rows of the matrix are associated with inputs and the columns are associated with states. In Klapper's cause-and-effects matrix, however, rows are associated with causes (inputs) and columns are associated with effects (output functions). The situation is reversed with respect to claims 5 and 22. Klapper is cited as a teaching matrix of cells wherein the columns are associated with inputs and the rows are associated with states. Again, the cells of Klapper's cause-and-effects matrix correspond to inputs and corresponding output functions, not inputs and corresponding states. In regard to claims 9 and 26 Klapper is cited as teaching that the number of cells is determined by the number of inputs. This statement is accurate with respect to Klapper's cause-and-effects matrix, but the cells in Klapper's cause-and-effects matrix do not

correspond to the cells claimed in claims 9 and 26. Finally, with respect to claim 47, Klapper is cited as teaching the retrieval of data based on the current state of the state machine and indicative of appropriate values for a plurality of state machine outputs. Klapper, however, does not disclose a state machine. The outputs set according to Klapper's cause-and-effects matrix are set based on the state of corresponding input signals, not the current state of a state machine.

As is clear from the above discussion, Klapper does not teach the features of dependent claims 2-5, 9, 19-22, 26 and 47 as suggested by the examiner. Furthermore, even if Klapper taught such features, the combination of Klapper with Kodosky would still fail to teach all of the features called for in the broader independent claims from which claims 2-5, 9, 19-22, 26 and 47 depend.

Next applicants turn to the rejection of claim 12-14, 28-30 and 52 as being unpatentable over Kodosky in view of Khrapunovich. Claims 12-14 depend from claim 1; claims 28-30 depend from claim 18; and claim 52 depends from claim 34. As has already been discussed, Kodosky fails to disclose all of the elements of independent claim 1, 18 and 34. For example, Kodosky fails to disclose a graphical user interface that defines a plurality of graphical elements defining state machine input/state pairs. Kodosky also fails to disclose receiving state transition data associated with one or more graphical elements identifying one or more next states to which a state machine is to transition following conditions in a process plant that correspond to the input/state pairs defined by the graphical elements of the graphical user interface. With regard to claims 12 and 29, Khrapunovich is cited as teaching receiving data indicative of how to handle inputs that have a BAD status, and storing the data

indicative of how to handle inputs that have a BAD status. Applicants take no position on whether Khrapunovich teaches such features, or whether Khrapunovich is properly combinable with Kodosky because, even if Khrapunovich does in fact teach the subject matter identified by the examiner, and even if the teaching of Khrapunovich may be combined with that of Kodosky, the combination still fails to disclose all of the claimed features of the base claims from which claims 12 and 24 depend. Therefore claims 12 and 29 are not unpatentable over the combination of Kodosky and Khrapunovich and should be allowed.

With regard to claims 13 and 28 Khrapunovich is cited as teaching receiving data indicative of priorities associated with a plurality of inputs and storing data indicative of how to handle inputs that have a BAD status. With regard to claims 14 and 30 Khrapunovich is cited as teaching data indicative of whether at least one input should be ignored, and storing the data indicative of whether to ignore at least one of the inputs. Finally, with regard to dependent claims 51 and 52 Khrapunovich is cited as teaching an instance where it would be desirable to force a signal to an initial state, setting the current state of the state machine to the initial state, and a single input is provided for indicating whether a function block is to be enabled or disabled. Again, even if Khrapunovich teaches these features, and even if Khrapunovich may be properly combined with Kodosky, the combination nonetheless fails to teach all of the features of the independent base claim from which claims 13, 14, 28, 29, 51 and 52 depend.

Next applicants turn to the rejection of claims 6-8, 23-25, 77 and 79. Claims 6-8 depend from claim 2, which itself depends from independent claim 1, and claims 23-25

depend from claim 19, which itself depends from claim 18. As has already been described, the combined teaching of Kodosky and Klapper fails to teach all of the elements of claims 2 and 19. With regard to claims 6-8 and 23-25 Khrapunovich is cited as teaching input signals that are in the form of zeros and ones. With regard to claims 6 and 23 Klapper is cited as teaching a plurality of cells associated with possible pairings of graphical elements with at least one input that is indicative of a next state. With regard to claims 7-8 and 24-25 Klapper is further cited as teaching a configuration tool that allows a user to define a number of cells based on a number of inputs.

Applicants note, however, that the matrix cells defined by Klapper do not correspond to input/state pairs, and do not receive and store data indicative of a next state to which a state machine transitions following conditions in a process plant that correspond to the input/state pair associated with one of the cells, as called for in the base claims from which claims 6-8 and 23-25 depend. Accordingly, even if Klapper and Khrapunovich teach the features cited by the examiner, the combination of Kodosky Klapper and Khrapunovich nonetheless fails to teach all of the features of dependent claims 6-8 and 23-25.

With regard to claims 77 and 79, Klapper is cited as teaching a matrix database stored on a memory. However, as discussed with regard to independent claim 58, from which claims 77 and 79 depend, the cause-and-effects matrix disclosed by Klapper does not correspond to a state machine configuration database as claimed in the present application. Thus, even if Kodosky, Klapper and Khrapunovich are properly combinable, they fail to teach all of the elements of base claim 58 and dependent claims 77 and 79.

For all of the reasons given above, claims 6-8, 23-25, 77, and 79 are not unpatentable over the combined teaching of Kodosky, Klapper and Khrapunovich.

Finally, applicants turn to the rejection of claims 10, 11, and 27 under 35 U.S.C. § 103(a) as being unpatentable over Kodosky, Klapper and Larson. Claims 10 and 11 depend from claim 2 which itself depends from claim 1. Claim 17 depends from claim 19 which itself depends from claim 18. As has already been discussed with regard to the rejection of claims 2 and 19 under 35 U.S.C. § 103(a) over Kodosky and Klapper, the combined teaching of Kodosky and Klapper fails to teach all of the elements of dependent claims 2 and 19. With regard to claims 10 and 11, and claim 27, Larson has been cited as teaching a second plurality of cells associated with function blocks, wherein the cells correspond to a plurality of outputs of the function block, and receiving data indicative of an output value. Applicants do not agree with the examiner's characterization of the teaching of Larson, however, even if Larson teaches the features ascribed to it by the examiner, the resultant combination of Kodosky, Klapper, and Khrapunovich nonetheless fails to teach all of the features of claims 2 and 19 from which claims 10, 11 and 27 depend. Accordingly, claims 10, 112 and 27 are not unpatentable under 35 U.S.C. § 103(a) over Kodosky in view of Klapper and further in view of Khrapunovich.

For all of the above reasons, the claims pending in the present application are allowable over the art of record. Applicants respectfully request the examiner to withdraw the rejections and allow the case to issue. If the examiner has any questions regarding the present response, he is encouraged to contact applicant's attorney at the number provided below.

Dated: June 13, 2008

Respectfully submitted,

By: /Jeffrey H. Canfield #38,404/
Jeffrey H. Canfield
Registration No.: 38,404
MARSHALL, GERSTEIN & BORUN LLP
233 S. Wacker Drive, Suite 6300
Sears Tower
Chicago, Illinois 60606-6357
(312) 474-6300
Attorney for Applicant